Attorney's Docket No. 6741.P089

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application for:

Claus Gschiermeister, et al.

Serial No. 10/562,380

Filed: December 23, 2005

For: **CHANGE NOTIFICATION AGENT**

Examiner: Kimbleann C. Verdi

Art Unit: 2194

Confirmation No.: 1914

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

The Appellants submit the following Appeal Brief pursuant to 37 C.F.R. § 41.37(c) for consideration by the Board of Patent Appeals and Interferences. The Appellants authorize the amount of $540.00 to cover the cost of filing the opening brief as required by 37 C.F.R. § 1.17(f) to be charged to Deposit Account No. 02-2666.

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

Claus Gschiermeister, Gabrial Alvarez, Dominic Poetschke and Grego Rieken, the parties named in the caption, transferred their rights to the subject Application through an assignment recorded on March 16, 2006 (Reel/Frame 017318/0098) in the patent application to SAP AG, of Walldorf, Germany. Thus, as the owner at the time the brief is being filed, SAP AG is the real party in interest.

II. RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences that will directly affect, be directly affected by or have a bearing on the Board's decision in this Appeal.

III. STATUS OF CLAIMS

Claims 1-28 are currently pending and rejected in the Application. The Appellants respectfully appeal the rejections of claims 1-28.

IV. STATUS OF AMENDMENTS

The Final Office Action was mailed on July 7, 2010. The Appellants filed their Response to the Final Office Action on August 24, 2010, wherein claims 1, 9, 14 and 20-28 were amended. The amended claims were entered as indicated in the Advisory Action mailed September 8, 2010.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

Claim 1 recites a computer-implemented method comprising: receiving a notification regarding a data object indicating a change to the data object (page 6 lines 30-31, FIG. 2 element 207); upon each receipt of the notification, requesting, by an agent executing in a computer system, changed data from the data object (page 6 lines 33-35, page 10 lines 24-30, FIG. 2 element 209); checking, by the agent, a plurality of entries representative of a plurality of applications maintained by the agent to determine whether the changed data is relevant for each application in the plurality of applications (page 7 lines 1-2, page 10 lines 32-35, FIG. 2 element 211), notifying, by the agent, an application about the changed data if the change is relevant for that application (page 7 lines 4-5, page 10 lines 20-22, FIG. 3 element 213); and transmitting, by the agent, the changed data to the application (page 7 lines 7-8, page 11 lines 10-11,FIG. 2 and FIG. 3 element 215).

Claim 7 recites the method of claim 1 further comprising: filtering out data from the changed data not to be communicated to the application, prior to transmitting the changed data to the application (page 8 lines 1-6, page 11 lines 5-8).

Claim 14 recites a computer system comprising: an agent executed by the computer system for administrating changes of data objects (page 6 lines 18-19, FIG. 1 and FIG. 4 element 100), the agent to register an entry representative of a data object in a first data structure (page 6 lines 25-26), to register an entry representative of an application in a second data structure (page 6 lines 26-27), the application entry specifying the data object whose changes are relevant for the application (page 6 lines 27-28), to call a first method by the agent to notify the application about a change to the data object (page 7 lines 4-5, page 10 lines 20-22, FIG. 3 element 213), to call a second method by the agent to obtain changed data from the data object (page 6 lines 33-35, page 10 lines 24-30, FIG. 2 element 209), to call a third method by the agent to check whether the change to the data object is relevant for the application (page 7 lines 1-2, page 10 lines 32-35, FIG. 2 element 211), and to call a fourth method by the agent to transmit relevant changed data to the application after notifying the application (page 7 lines 7-8, page 11 lines 10-11,FIG. 2 and FIG. 3 element 215).

Claim 15 recites the computer system of claim 14, wherein the agent generates a first input interface to allow for registering the entry representative of the data object (page 8 lines 8-13, page 12 lines 13-19, FIG. 5 element 530), and generates a second input interface to allow for registering the entry representative of the application (page 8 lines 8-13, page 12 lines 13-19, FIG. 5 element 550).

Claim 21 recites computer system of claim 14, wherein the agent filters out data from the relevant changed data that is not to be communicated to the application, prior to transmitting the relevant changed data to the application (page 8 lines 1-6, page 11 lines 5-8).

Claim 28 recites non-transitory machine readable storage medium having instructions stored therein, the instructions when executed by a processor cause the machine to perform a set of operations (page 11 lines 15-20) comprising: storing a data object entry in a first data structure (page 6 lines 25-26); storing an application entry in a second data structure (page 6 lines 26-27), the application entry specifying the data object whose changes are relevant for an application (page 6 lines 27-28); receiving a notification regarding the data object as to a change to the data object (page 6 lines 30-31, FIG. 2 element 207); upon receipt of the notification, getting changed data from the data object (page 6 lines 33-35, page 10 lines 24-30, FIG. 2 element 209); checking among a plurality of applications whether the change is relevant for each application in the plurality of applications (page 7 lines 1-2, page 10 lines 32-35, FIG. 2 element 211), notifying the application about the change if the change is relevant for the application (page 7 lines 4-5, page 10 lines 20-22, FIG. 3 element 213); and transmitting the relevant changed data to the application (page 7 lines 7-8, page 11 lines 10-11,FIG. 2 and FIG. 3 element 215).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1, 4, 5, 7, 9, 14, 15, 18, 19, 21, 23, and 28 are rejected under 35 U.S.C. §103(a) as allegedly being obvious over U.S. Patent No. 5,991,536 by Brodsky, *et al.* (hereinafter "Brodsky") in view of U.S. Patent No. 6,029,175 by Chow, *et al.* (hereinafter "Chow").

Claims 6, 8, 13, 20, 22, and 27 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Brodsky in view of Chow and in further view of U.S. Publication No. 2005/0015441 by Attwood, *et al.* (hereinafter "Attwood").

Claims 2, 3, 10-12, 16, 17, and 24-26 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Brodsky in view of Chow, as applied to claims 1 and 14, and further in view of U.S. Patent No. 6,044,205 by Reed, *et al.* (hereinafter "Reed").

All of the claims do not stand or fall together. The basis for separate patentability of the claims is set forth below.

VII. ARGUMENT

A. Rejection of Claims 1, 4, 5, 7, 9, 14, 15, 18, 19, 21, 23 and 28 Under 35 U.S.C. §103(a) based on Brodsky and Chow

Claims 1, 4, 5, 7, 9, 14, 15, 18, 19, 21, 23, and 28 are rejected under 35 U.S.C. §103(a) as allegedly being obvious over Brodsky in view of Chow. To establish a *prima facie* case of obviousness, the Examiner must provide some articulated reasoning to support the conclusion of obviousness. KSR International Co. v. Teleflex Inc., 127 S.Ct. 1727, 1741, 82 USPQ2d 1385 (2007) (quoting In re Kahn, 441 F.3d 977, 988, USPQ2d 1329, 1336 (Fed. Cir. 2006)).

1. Independent Claims 1, 14, and 28; Dependent Claims 2-6, 8-12, 16-20, and 22-26

a) Independent claims 1, 14, and 28 and Dependent Claims 2-6, 8-12, 16-20, and 22-26 are not obvious at least because Brodsky and Chow fail to teach requesting changed data from the data object

Independent claim 1 recites "requesting, by an agent executing in a computer system, changed data from the data object." Independent claim 14 recites "to call a second method by the agent to obtain changed data from the data object." Independent claim 28 recites "getting changed data from the data object." The Examiner concedes that Brodsky does not explicitly disclose this limitation. However, the Examiner alleges that Chow teaches this limitation. See Final Office Action mailed July 7, 2010, pages 6-7; See Advisory Action mailed September 8, 2010, pages 2-3.

The Examiner cites step 459 of figure 38 and column 29 lines 50-57 of Chow as teaching "request for the object." See Final Office Action mailed July 7, 2010, pages 6-7; See Advisory Action mailed September 8, 2010, page 2. Column 29 lines 50-57 of Chow describes step 459 of figure 38 and states the following:

> If a search specification is defined for the object, then the object is obtained from the network in step 459 by following the object's search specification. For example, the object identification code itself may specify a primary or unique source for the object, and in this case the Revision Manager directs a request over the network to the primary or unique source for the object.

See Chow, column 29 lines 50-57. However, this section of Chow does not disclose that the Revision Manager requests or obtains changed data from the object. Rather, Chow discloses that the entire object is obtained from the network. Furthermore, the clients in Chow issue a GET command to the Revision Manager to access the object which causes the clients to reload the entire object. See Chow, column 6 lines 7-15. Therefore, this section of Chow does not teach requesting or getting changed data from the data object

The Examiner also cites column 30 lines 35-57 of Chow as teaching "the object obtained from the source contains the changed data of the object." See Advisory Action mailed September 8, 2010, page 2. The relevant portions of the cited section state the following:

> Once the object has been obtained . . . the change in the object is determined by comparing the new version of the object to the existing version in the cache, and the change is logged in the log of changes to the object (386 in FIG. 34) along with a time stamp. Then in step 464 the object is updated and time stamped in the cache. After step 464, the change in the object is evaluated by any significant change detection method defined for the object (389 in FIG. 34).

See Chow, column 30 lines 35-50; See Advisory Action mailed September 8, 2010, page 2. However, this section of Chow does not teach requesting or obtaining changed data from the object. Rather, Chow discloses that the entire object is obtained and then the change in the object is determined and evaluated to detect any significant change. If the change is significant to a client, the Revision Manager notifies the client which causes the client to reload the entire object. See Chow, column 32 lines 19-58. Therefore, this section of Chow does not teach requesting or getting changed data from the data object.

The Examiner asserts that Chow represents requesting changed data from the data object because the object obtained by the Revision Manager contains the changed data. See Advisory Action mailed September 8, 2010, pages 2-3. However, obtaining an object that contains changed data does not necessarily infer or imply that changed data was requested from the data object. Rather, as disclosed in Chow, when the Revision Manager receives a change notification, the Revision Manager obtains the entire object from the network without specifically requesting the changed data from the object. See Chow, column 29 lines 20-59. It is after the object is obtained that the Revision Manager determines the change in the object by comparing the new version of the object with the existing version in the cache. See Chow, column 30 lines 35-57. Similarly, when a client receives a change notification, the client reloads the entire object without specifically requesting the changed data from the object. See Chow, column 32, lines 41-58. The client then identifies the changes in the object by comparing the new version of the object with the old version in the storage. Id. Therefore, the Examiner's assertion is not supported by Chow.

The Examiner further states the following:

> It would have been obvious to a person of ordinary skill in the art at the time the invention was made to have modified the notification manager of Brodsky with the teachings of Revision Manager from Chow because this feature would have provided a mechanism which collects at a site convenient to a group of users a single cache of the most recent versions of documents so that all members of the group can have quick and inexpensive access, while the group as a whole can significantly reduce communication costs (col. 6, lines 57-62 of Chow).

See Advisory Action mailed September 8, 2010, page 2. However, the Examiner has not articulated "a finding that there was some teaching, suggestion, or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings" as required by M.P.E.P. §2143. Brodsky discloses that a server maintains the object hierarchy. See Brodsky, column 5 lines 64-65. The object hierarchy includes the observed objects, and when a change is made to an observed object, this change is reflected in the object hierarchy. Id. at column 3 lines 31-34. The observer objects can access and interact with the object hierarchy to obtain information about the observed objects. Id. at column 3 lines 36-38, column 4 lines 16-18 and lines 37-39. Therefore, there is no motivation to modify the notification manager of Brodsky with the teachings of the revision manager of Chow to provide a mechanism which collects at a site the most recent versions of documents because Brodsky provides for this mechanism via its object hierarchy.

For at least the reasons mentioned above, the Examiner has failed to establish a *prima facie* case of obviousness because the Examiner has not established that Chow teaches "requesting, by an agent executing in a computer system, changed data from the data object," as recited in claim 1, "to call a second method by the agent to obtain changed data from the data object," as recited in claim 14, and "getting changed data from the data object," as recited in claim 28. Thus, the Examiner has not established that Brodsky and Chow teach each element of claims 1, 14, and 28. Furthermore, the Examiner has not articulated a finding that there was some teaching, suggestion, or motivation to modify Brodsky with the teachings of Chow. Accordingly, the Appellants respectfully request that the rejection of claims 1, 14, and 28 be reversed.

Dependent claims 2-6, 8-12, 16-20, and 22-26 depend from one of independent claims 1 and 14 and thus incorporate the respective limitations thereof. For at least the aforementioned reasons relating to independent claims 1 and 14, Brodsky and Chow fail teach each and every element of these dependent claims. Accordingly, the Appellants respectfully request that the rejection of claims 2-6, 8-12, 16-20, and 22-26 be reversed.

2. Claims 7 and 21

a) Claims 7 and 21 depend from patentable base claims 1 and 14

Claims 7 and 21 depend from independent claims 1 and 14, respectively, and thus incorporate the limitations thereof. For at least the aforementioned reasons regarding claims 1 and 14, Brodsky and Chow fail teach each and every element of claims 7 and 21. Accordingly, the Appellants respectfully request that the rejection of claims 7 and 21 be reversed. Further, the Appellants believe that these claims are separately patentable for the reasons below.

b) Claims 7 and 21 are not obvious because Brodsky and Chow fail to teach filtering out data from the changed data not to be communicated to the application

Claim 7 recites "filtering out data from the changed data not to be communicated to the application, prior to transmitting the changed data to the application." Claim 21 recites "the agent filters out data from the relevant changed data that is not to be communicated to the application, prior to transmitting the changed data to the application." The Examiner alleges that Brodsky teaches "filtering out data objects whose changes are not to be communicated to an application" and cites column 4 lines 28-30, lines 45-46, and lines 51-57 of Brodsky. See Final Office Action mailed July 7, 2010, page 9.

The cited sections of Brodsky disclose that the observer objects register with the notification manager so that they are notified when changes or modifications are made to the observed objects in the object hierarchy, where each observed object in the object hierarchy may have multiple observer objects. See Brodsky, column 4 lines 28-31 and lines 44-47. The notification manager interfaces with the object hierarchy, accesses certain status information contained in the object hierarchy, determines which observed objects have been modified, and then notifies the associated observer objects with information concerning the changes. Id., column 4 lines 51-57.

However, the Examiner's interpretation of Brodsky is improper. The Examiner states that the above sections of Brodsky teach the elements of claims 7 and 21 because the "notification manager filters out observed objects that do not have associated observed [sic] objects since the notification manager only sends notifications to registered observer objects listed with the observed object." See Final Office Action mailed July 7, 2010, page 9. However, it does not follow that because the notification manager only sends notifications to registered observer objects associated with the modified observed objects, the notification manager filters out observed objects that do not have associated observer objects. By definition, an observed object must have an associated observer object. Otherwise, the object would not be an observed object. Therefore, the notification manager does not filter out observed objects that do not have associated observer objects because every one of the observed objects has at least one associated observer object. See Brodsky, column 4 lines 44-47.

The cited sections of Brodsky actually set forth that the notification manager determines which observed objects have been changed and sends notifications to the observer objects associated with the modified observed objects with information concerning the changes. See Brodsky, column 4 lines 44-47 and lines 51-57. However, Brodsky does not teach that the notification manager filters out data from the relevant changed data that is not to be communicated to the observer object, prior to transmitting the relevant changed data to the observer object. In Appellants' claims, the changed data refers to "changed data from the data object," as recited in independent claims 1 and 14. Brodsky does not mention that the notification manager accesses the data of the observed object, much less that the notification manager filters any data from the observed object. Rather, Brodsky discloses that the notification manager interfaces to the object hierarchy and accesses certain status information contained in the object hierarchy. See Brodsky, column 4 lines 51-55. From this status information, the notification manager determines which observed objects have been modified and then notifies the associated observer objects with information concerning the changes. Id. Thus, the notification manager never filters, or even accesses, the data of the modified observed object, but rather only interfaces with the object hierarchy to determine which observed objects have been changed.

Therefore, Brodsky fails to teach the above limitation of claims 7 and 21. The Examiner does not indicate and the Appellants' cannot discern any part of Chow that cures the aforementioned deficiencies of Brodsky. Thus, the Examiner has not established that Brodsky and Chow teach each element of claims 7 and 21 as required to establish a *prima facie* case of obviousness. Accordingly, the Appellants respectfully request that the rejection of claims 7 and 21 be reversed.

3. Claim 15

a) Claim 15 depends from patentable base claim 14

Claim 15 depends from independent claim 14 and thus incorporates the limitations thereof. For at least the above reasons regarding claim 14, Brodsky and Chow fail to teach each and every element of claim 15. Accordingly, the Appellants respectfully request that the rejection of claim 15 be reversed. Further, the Appellants believe that this claim is separately patentable for the reasons below.

b) Claim 15 is not obvious because Brodsky and Chow fail to teach the agent generating a second input interface to allow for registering the entry representative of the application

Claim 15 recites "the agent generates a first input interface to allow for registering the entry representative of the data object, and the agent generates a second input interface to allow for registering the entry representative of the application." The Examiner alleges that Brodsky teaches these elements and cites column 3 lines 60-67 and column 4 lines 1-2 and 28-29 of Brodsky. See Office Action mailed July 7, 2010, page 12.

The cited sections of Brodsky disclose that a typical observer object is a user interface object such as a view or window displayed on a monitor attached to a computer, which registers with the notification manager to be notified when changes are made to an observed object. See Brodsky, column 3 lines 60-63, column 4 lines 28-31. The views are used to interact with the user in the construction of an object hierarchy or its component parts. Id., column 3 line 64-column 4line 7. Each of the views may comprise one or more observer objects or a component part of a single observer object. Id., column 4 lines 8-10.

However, Brodsky does not teach that the notification manager generates a first input interface and a second input interface, where the first input interface allows for registering an observed object and the second input interface allows for registering an observer object. Rather, Brodsky discloses a number of different views to allow the user to construct an object hierarchy or its component parts, which may comprise one or more observer objects or a component part of a single observer object. See Brodsky, column 3 line 64-column 4 line 10. Thus, Brodsky does not teach generating a view for registering an observed object and generating a different view for registering an observer object.

Therefore, Brodsky fails to teach the above limitation of claim 15. The Examiner does not indicate and the Appellants' cannot discern any part of Chow that cures the aforementioned deficiencies of Brodsky. Thus, the Examiner has not established that Brodsky and Chow teach each element of claim 15 as required to establish a *prima facie* case of obviousness. Accordingly, the Appellants respectfully request that the rejection of claim 15 be reversed.

B. Rejection of Claims 6, 8, 13, 20, 22, and 27 Under 35 U.S.C. §103(a) based on Brodsky, Chow, and Attwood

Claims 6, 8, 13, 20, 22, and 27 are rejected under 35 U.S.C. § 103(a) as allegedly being obvious over Brodsky in view of Chow and further in view of U.S. Publication No. 2005/0015441 by Attwood, *et al.* (hereinafter "Attwood"). To establish a *prima facie* case of obviousness, the Examiner must provide some articulated reasoning to support the conclusion of obviousness. KSR International Co. v. Teleflex Inc., 127 S.Ct. 1727, 1741, 82 USPQ2d 1385 (2007) (quoting In re Kahn, 441 F.3d 977, 988, USPQ2d 1329, 1336 (Fed. Cir. 2006)).

Claims 6, 8, 13, 20, 22, and 27 depend from one of independent claims 1 and 14 and thus incorporate the limitations thereof. The Examiner does not indicate and the Appellants do not discern any part of Attwood that cures the aforementioned deficiencies of Brodsky and Chow set forth above. For at least the aforementioned reasons regarding independent claims 1 and 14, Brodsky, Chow, and Attwood do not teach or suggest all the limitations of these dependent claims. Accordingly, the Appellants respectfully request that the rejection of claims 6, 8, 13, 20, 22, and 27 be reversed.

C. Rejection of Claims 2, 3, 10-12, 16, 17, and 24-26 Under 35 U.S.C. §103(a) based on Brodsky, Chow, and Reed

Claims 2, 3, 10-12, 16, 17 and 24-26 are rejected under 35 U.S.C. § 103(a) as allegedly being obvious over Brodsky in view of Chow, as applied to claims 1 and 14, and further in view of U.S. Patent No. 6,044,205 by Reed, *et al.* (hereinafter "Reed"). To establish a *prima facie* case of obviousness, the Examiner must provide some articulated reasoning to support the conclusion of obviousness. KSR International Co. v. Teleflex Inc., 127 S.Ct. 1727, 1741, 82 USPQ2d 1385 (2007) (quoting In re Kahn, 441 F.3d 977, 988, USPQ2d 1329, 1336 (Fed. Cir. 2006)).

Claims 2, 3, 10-12, 16, 17, and 24-26 depend from one of independent claims 1 and 14 and thus incorporate the limitations thereof. The Examiner does not indicate and the Appellants do not discern any part of Reed that cures the aforementioned deficiencies of Brodsky and Chow. For at least the above reasons regarding independent claims 1 and 14, Brodsky, Chow, and Reed do not teach or suggest all the limitations of these dependent claims. Accordingly, the Appellants respectfully request that the rejection of claims 2, 3, 10-12, 16, 17, and 24-26 be reversed.

CONCLUSION

In view of the foregoing, the Appellants respectfully request reversal of all the above rejections in connection with claims 1-28.

Respectfully submitted,

BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP

Dated: November 5, 2010

/Thomas M. Coester/
Thomas M. Coester, Reg. No. 39,637

1279 Oakmead Parkway
Sunnyvale, California 94085-4040
Telephone (310) 207-3800
Facsimile (310) 820-5270

CERTIFICATE OF TRANSMISSION

I hereby certify that this correspondence is being submitted to the United States Patent and Trademark Office electronically via EFS Web on the date shown below.

Marilyn Bass

Marilyn Bass November 10, 2010

VIII. CLAIMS APPENDIX

1. (Previously Presented) A computer-implemented method comprising:

receiving a notification regarding a data object indicating a change to the data object;

upon each receipt of the notification, requesting, by an agent executing in a computer system, changed data from the data object;

checking, by the agent, a plurality of entries representative of a plurality of applications maintained by the agent to determine whether the changed data is relevant for each application in the plurality of applications,

notifying, by the agent, an application about the changed data if the change is relevant for that application; and

transmitting, by the agent, the changed data to the application.

2. (Previously Presented) The method of claim 1, further comprising:

receiving a confirmation from the application after transmitting the changed data to the application.

3. (Previously Presented) The method of claim 2, further comprising:

triggering a mechanism if the confirmation of the changed data is not received from the application.

4. (Previously Presented) The method of claim 1, further comprising:

registering an entry of a sub-object, the sub-object being a set of data which is changed in dependence on the change to the data object.

5. (Previously Presented) The method of claim 4, further comprising:

transmitting changed sub-object data to the application after notifying the application.

6. (Previously Presented) The method of claim 1, further comprising:

determining a list of fields with changes that are relevant for the application.

7. (Previously Presented) The method of claim 1 further comprising:

filtering out data from the changed data not to be communicated to the application, prior to transmitting the changed data to the application.

8. (Previously Presented) The method of claim 1, further comprising:

registering an entry representative of the application; and

specifying a type of change to the data object that is relevant for the application.

9. (Previously Presented) The method of claim 1 further comprising:

registering an entry for the data object and the application in a customization structure of the agent.

10. (Previously Presented) The method of claim 9, wherein the entry for the data object comprises:

- an ID representative of the data object;
- an ID representative of a key of the data object;
- a flag representative of an activity;
- an ID representative of a key structure of the data object; and
- an ID of a wrapper class.
-

11. (Previously Presented) The method of claim 9, wherein the entry for the application comprises:

- an ID representative of the application;
- a flag representative of an activity; and
- an ID representative of an expected structure of notification.

12. (Previously Presented) The method of claim 4, wherein the entry for the sub-object comprises:

- an ID representative of the sub-object;
- an ID representative of a key data object;

- an ID representative of a structure of the data object; and
- an ID representative of an object key object.

13. (Previously Presented) The method of claim 1, wherein the data object represents one of a location, a location-product, and a transportation lane in a context of a business application.

14. (Previously Presented A computer system comprising:

an agent executed by the computer system for administrating changes of data objects, the agent to register an entry representative of a data object in a first data structure, to register an entry representative of an application in a second data structure, the application entry specifying the data object whose changes are relevant for the application, to call a first method by the agent to notify the application about a change to the data object, to call a second method by the agent to obtain changed data from the data object, to call a third method by the agent to check whether the change to the data object is relevant for the application, and to call a fourth method by the agent to transmit relevant changed data to the application after notifying the application.

15. (Previously Presented) The computer system of claim 14, wherein the agent generates a first input interface to allow for registering the entry representative of the data object, and generates a second input interface to allow for registering the entry representative of the application.

16. (Previously Presented) The computer system of claim 14, wherein the agent receives a confirmation from the application after transmitting the relevant changed data to the application.

17. (Previously Presented) The computer system of claim 16, wherein the agent triggers a mechanism if the confirmation is not received from the application.

18. (Previously Presented) The computer system of claim 14, wherein the agent registers an entry of a sub-object, the sub-object being a set of data which is changed in dependence on a change of a key data object.

19. (Previously Presented) The computer system of claim 18, wherein the agent transmits a relevant change to the sub-object to the application after notifying the application.

20. (Previously Presented) The computer system of claim 14, wherein the agent maintains a list of fields whose changes are relevant for the application.

21. (Previously Presented) The computer system of claim 14, wherein the agent filters out data from the relevant changed data that is not to be communicated to the application, prior to transmitting the relevant changed data to the application.

22. (Previously Presented) The computer system of claim 14, wherein the entry representative of the application specifies which changes of the data object are relevant for the application.

23. (Previously Presented) The computer system of claim 14, wherein the entry of the data object and the application are registered in a customization structure of the agent.

24. (Previously Presented) The computer system of claim 14, wherein an entry for the data object comprises:

- an ID representative of the data object;
- an ID representative of a key of the data object;
- a flag representative of an activity;
- an ID representative of a key structure of the data object; and
- an ID of a wrapper class.

25. (Previously Presented) The computer system of claim 14, wherein the entry for the application comprises:

- an ID representative of the application;
- a flag representative of an activity; and
- an ID representative of an expected structure of notification.

26. (Previously Presented) The computer system of claim 14, wherein an entry for a sub-object comprises:

- an ID representative of the sub-object;
- an ID representative of a key data object;
- an ID representative of a structure of the data object; and
- an ID representative of an object key object.

27. (Previously Presented) The computer system of claim 14, wherein the data object represents one of a location, a location-product, and a transportation lane in a context of a business application.

28. (Previously Presented) A non-transitory machine readable storage medium having instructions stored therein, the instructions when executed by a processor cause the machine to perform a set of operations comprising:

storing a data object entry in a first data structure;

storing an application entry in a second data structure, the application entry specifying the data object whose changes are relevant for an application;

receiving a notification regarding the data object as to a change to the data object;

upon receipt of the notification, getting changed data from the data object;

checking among a plurality of applications whether the change is relevant for each application in the plurality of applications,

notifying the application about the change if the change is relevant for the application; and

transmitting the relevant changed data to the application.

IX. EVIDENCE APPENDIX

No evidence is submitted with this appeal.

X. RELATED PROCEEDINGS APPENDIX

No related proceedings exist.